

PATENT**Serial No. 09/506,640****Attorney Docket No. 1999P07475US01 (1009-023)****REMARKS**

The Examiner is respectfully thanked for the thoughtful consideration provided to this application. Reconsideration of this application is respectfully requested in light of the foregoing amendments and the following remarks.

Each of claims 59 and 60 has been amended solely for either addressing an informality or correcting a typographical error, thus explicitly presenting one or more elements implicit in the claim as originally written when viewed in light of the specification, and therefore not changing the scope of the claim.

Each of claims 1-25, 27, 28, 30-40, 42, 43, and 45-62 is pending in this application. Claims 1, 13, 21, 36, and 51 are in independent form.

I. The Rejections Under 35 U.S.C. § 112

Each of claims 59 and 60 was rejected under 35 U.S.C. 112, second paragraph, as lacking sufficient antecedent basis for the phrase "The method of claim 21" in line a.

Each of claims 59 and 60 has been amended solely for either addressing an informality or correcting a typographical error, thus explicitly presenting one or more elements implicit in the claim as originally written when viewed in light of the specification, and therefore not changing the scope of the claim. Thus, reconsideration and withdrawal of these rejections is respectfully requested.

II. The Obviousness Rejections

Claims 1-25, 27, 28, 30-40, 42, 43, and 45-62 were rejected under 35 U.S.C. § 103(a) as being unpatentable over various combinations of Koppolu (U.S. Patent No. 6,460,058) in view of Bonet (U.S. Patent No. 6,564,242). These rejections are respectfully traversed.

None of the cited references, either alone or in any combination, establishes a *prima facie* case of obviousness. "To establish a *prima facie* case of obviousness, three basic criteria must be

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure." *See* MPEP § 2143. Moreover, the USPTO "has the initial duty of supplying the factual basis for its rejection." *In re Warner*, 379 F.2d 1011, 154 USPQ 173, 178 (C.C.P.A. 1967).

A. The Present Office Action Misinterprets Koppolu

The present Office Action purports to find matter in Koppolu that is not present therein.

Claim 1 recites, yet Koppolu fails to expressly or inherently teach or suggest, "a specific instance associated with a **programmable logic controller comprising a first operating system**, wherein said programmable logic controller is communicatively coupled to a server running under a **second operating system on a personal computer**."

Claim 13 recites, yet Koppolu fails to expressly or inherently teach or suggest, "a specific instance of an **object associated with a first operating system**, wherein said specific instance is **not registered with a second operating system** such that a server running under said second operating system is not able to normally access said specific instance."

Claim 21 recites, yet Koppolu fails to expressly or inherently teach or suggest, "a memory adapted to store a specific instance, said memory associated with a **programmable logic controller that comprises a first operating system**" and "via a moniker, provide access to said specific instance via a **personal computer comprising a second operating system** such that said specific instance of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system."

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

Claim 36 recites, yet Koppolu fails to expressly or inherently teach or suggest, “memory means adapted to store a specific instance, said memory associated with a **programmable logic controller that comprises a first operating system**” and “via a moniker, provide access to said specific instance via a **personal computer comprising a second operating system** such that said specific instance of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system.”

Claim 51 recites, yet Koppolu fails to expressly or inherently teach or suggest, “for a specific instance of an object associated with a **first operating system**, wherein said specific instance is **not registered with said second operating system** such that a server of said second operating system is not able to normally access said specific instance.”

The present Office Action recites “for a specific instance (instance data structure of an object, lines 9-11 column 11) comprising a first operating system, line 58 column 11) communicates to a server running under a **second operating system** on a personal computer (**operating system running under server 100, Fig. 5**).” See Page 3. This same reasoning is also referenced in rejecting each of independent claims 13, 21, 36, and 51. See Pages 5-8. Applicant respectfully traverses this assertion as factually erroneous.

Koppolu defines the “server” allegedly illustrated in Fig. 5 by allegedly reciting “[r]eferring now to FIG. 5, the virtual function table 84 and member functions 86-88 of the object 80 are provided by a **server application program 100** which is **stored in the computer system 20 (FIG. 1) as an executable program file** (with a “.exe” file name extension) or as a **dynamic link library file** (with a “.dll” file name extension).” See col. 11, lines 51-56.

Where does Koppolu teach or recite that the “server application program” “stored” “as an executable program file” or as a “dynamic link library file” is associated with a **second operating system** as asserted in the present Office Action?

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

Instead, the processes described in Koppolu relating to Fig. 5 take place within a single “computer.” See e.g., col. 15, lines 1-14. Koppolu recites only a single operating system associated with this “computer”, as illustrated in Fig. 1. See e.g., col. 7, line 64 – col. 8, line 1.

Bonet fails to overcome the deficiencies of Koppolu.

B. Additional Argument – Independent Claims

Claim 1 recites, yet Koppolu fails to expressly or inherently teach or suggest, “determining that **said specific instance is not registered** with said second operating system and that **said specific instance is already running** under said first operating system; and utilizing a moniker, automatically **registering said specific instance** with said second operating system, such that said specific instance, is accessible by said server by checking a running object table of said second operating system.”

Claim 13 recites, yet Koppolu fails to expressly or inherently teach or suggest, “determining that **said specific instance is not registered** in a running object table of said second operating system and that **said specific instance is already running** under said first operating system; and utilizing a moniker, **registering said specific instance** with said second operating system such that said specific instance of said object, that was previously not registered with said second operating system such that said server was not able to normally access said specific instance, is accessible by said server by checking said running object table of said second operating system.”

Claim 21 recites, yet Koppolu fails to expressly or inherently teach or suggest, “a memory adapted to store a specific instance, said memory associated with a programmable logic controller that comprises a first operating system” and “**register said specific instance with said first operating system**; and via a moniker, **provide access to said specific instance via a personal computer comprising a second operating system such that said specific instance**

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system.”

Claim 36 recites, yet Koppolu fails to expressly or inherently teach or suggest, “register said specific instance with said first operating system; and via a moniker, provide access to said specific instance via a personal computer comprising a second operating system such that said specific instance of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system.”

Claim 51 recites, yet Koppolu fails to expressly or inherently teach or suggest, “determining that said specific instance is not registered in a running object table of said second operating system and that said specific instance is already running under said first operating system; and utilizing a moniker, registering said specific instance with said second operating system such that said specific instance of said object, that was previously not registered with said second operating system such that said server was not able to normally access said specific instance, is accessible by said server by checking said running object table of said second operating system.”

Koppolu allegedly recites when “the moniker 120 can look in the running objects table ... [I]f the object exists, the moniker 120 can simply return an interface pointer of the existing object to the client 104, and thus avoid creating the object again”. See col. 15 lines 60-63. Thus, Koppolu creates objects referenced by “moniker 120” that are not present in “the running objects table,” but does not expressly or inherently teach or suggest “registering” instances that are already “running.”

In response to this previously presented argument, the present Office Action claims, regarding Koppolu, that “the reference describes the process of checking already running objects of whether they are already registered with the system or not. If they are, the system does not need to register the objects again (lines 60-66 column 15).” See Page 8.

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

Instead of the characterization asserted in the present Office Action, the referenced lines in Koppolu actually recite that “[w]hen called by the client to bind to the named object 80 for example, the moniker 120 can look in the running objects table of the bind context 126 to determine whether the object already exists. If the object exists, the moniker 120 can simply **return an interface pointer of the existing object to the client 104, and thus avoid creating the object again.** Other uses of the bind context are described in *Inside OLE, second edition*, supra.”

Where does Koppolu expressly or inherently teach or suggest the claimed elements for independent claims 1, 13, 21, 36, and 51? For example, regarding claim 1, where does Koppolu expressly or inherently teach or suggest “determining that **said specific instance is not registered** in a running object table of said second operating system and that said specific instance **is already running** under said first operating system; and utilizing a moniker, **registering said specific instance** with said second operating system such that said specific instance of said object, that was previously not registered with said second operating system such that said server was not able to normally access said specific instance, is accessible by said server by checking said running object table of said second operating system”?

Koppolu does not expressly or inherently teach or suggest all of the elements of independent claims 1, 13, 21, 36, and 51. Bonet does not overcome the deficiencies of Koppolu.

C. Claim 2

Claim 2 recites, “said step of registering does not instantiate objects that are not running such that a dormant programmable logic controller is not erroneously activated.”

The present Office Action recites that “Koppolu as modified further teaches registering does not instantiate objects that are not running (254-259, Fig. 8A). See Page 4. Applicant respectfully traverses this contention as factually erroneous.

Fig. 8A of Koppolu allegedly illustrates the following flowchart steps:

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

254 – “object is running?”

255 – if “object is running”, “retrieve object”

256 – if “object” is not “running” “get bind information from client’s bind status callback

257 – if “object” is not “running” “initiate binding with transport”

258 – if “object” is not “running” “pass transport’s IBINDING to client via

IBINDSTATUSCALLBACK::ONSTARTBINDING

259 – if “object” is not “running” “return MK_S_ASYNCHRONOUS and null
object pointer from IMONIKER::BINDTOOBJECT.

Regarding “BINDTOOBJECT” function, Koppolu allegedly recites, “[i]n the moniker's **BindToObject** function, the moniker 120 includes code which launches the object's server application program 100, requests creation of the named object, and queries for a pointer to the interface requested by the client. *See* col. 16, lines 40-43.

How do these steps or Koppolu as modified expressly or inherently teach or suggest, “said step of registering **does not instantiate objects that are not running** such that a dormant programmable logic controller is not erroneously activated”?

D. Suggestion or Motivation to Combine Koppolu with Bonet

“When determining the patentability of a claimed invention which combines two known elements, the question is whether there is something in the prior art as a whole to **suggest the desirability**, and thus the obviousness, of making the combination.” *Akamai Techs. v. Cable & Wireless Internet Servs.*, 344 F.3d 1186, 68 USPQ 2d 1186 (Fed. Cir. 2003) (emphasis added).

According to the Federal Circuit the “**mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification.**” *In re Fritch*, 972 F.2d 1260, 23 USPQ 2d 1780, 1783-784 (Fed. Cir. 1992) (quoting *In re Fine*, 837 F.2d 1071, 1075, 5 USPQ 2d 1596, 1600 (Fed. Cir. 1988)). In that same case, the Federal Circuit further held that it “is

PATENT**Serial No. 09/506,640****Attorney Docket No. 1999P07475US01 (1009-023)**

impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious. This court has previously stated that "[o]ne cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention." *Id.*, 23 USPQ 2d at 1784.

Instead of meeting this standard, the present Office Action recites "[i]t would have been obvious to apply the teachings of Bonet to the system of Koppolu because the specific instance of Koppolu **could be associated** with a programmable logic controller wherein this specific instance is being used by the client program to control the operation of a distributed automation system as disclosed by Bonet (lines 41-53 column1)." *See* Page 4.

Even if this statement from the operation were correct, an assumption with which Applicant disagrees, it facially fails to provide any evidence whatsoever regarding a suggestion of "the desirability of the modification" as required by law.

E. Conclusion – Nonobviousness

Thus, even if there were motivation or suggestion to modify or combine the cited references (an assumption with which Applicant disagrees), and even if there were a reasonable expectation of success in combining or modify the cited references (another assumption with which Applicant disagrees), the cited references still do not expressly or inherently teach or suggest every limitation of the independent claims, and consequently fail to establish a *prima facie* case of obviousness.

Because no *prima facie* rejection of any independent claim has been presented, no *prima facie* rejection of any dependent claim can be properly asserted. Consequently, reconsideration and withdrawal of these rejections is respectfully requested.

PATENT**Serial No. 09/506,640****Attorney Docket No. 1999P07475US01 (1009-023)****II. Allowable Subject Matter**

The following is a statement of reasons for the indication of allowable subject matter:

“none of the references of record alone or in combination disclose or suggest the combination of limitations found in the independent claims. Namely:

claims 1-12 are allowable because none of the references of record alone or in combination disclose or suggest ‘determining that said specific instance is not registered with said second operating system and that said specific instance is already running under said first operating system; and utilizing a moniker, automatically registering said specific instance with said second operating system’;

claims 13-20 are allowable because none of the references of record alone or in combination disclose or suggest ‘determining that said specific instance is not registered in a running object table of said second operating system and that said specific instance is already running under said first operating system; and utilizing a moniker, registering said specific instance with said second operating system’;

claims 21-25, 27, 28, 30-35, 59, and 60 are allowable because none of the references of record alone or in combination disclose or suggest ‘a processor adapted to: register said specific instance with said first operating system; and via a moniker, provide access to said specific instance via a personal computer comprising a second operating system such that said specific instance of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system’;

claims 36-40, 42, 43, 45-50, 61, and 62 are allowable because none of the references of record alone or in combination disclose or suggest ‘processor means adapted to: register said specific instance with said first operating system; and via a moniker, provide access to said specific instance via a personal computer

PATENT**Serial No. 09/506,640****Attorney Docket No. 1999P07475US01 (1009-023)**

comprising a second operating system such that said specific instance of said object is accessible by a server associated with said personal computer by accessing a running objects table of said second operating system'; and

claims 51-58 are each allowable because none of the references of record alone or in combination disclose or suggest 'determining that said specific instance is not registered in a running object table of said second operating system and that said specific instance is already running under said first operating system; and utilizing a moniker, registering said specific instance with said second operating system.'"

PATENT

Serial No. 09/506,640

Attorney Docket No. 1999P07475US01 (1009-023)

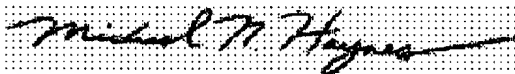
CONCLUSION

It is respectfully submitted that, in view of the foregoing amendments and remarks, the application as amended is in clear condition for allowance. Reconsideration, withdrawal of all grounds of rejection, and issuance of a Notice of Allowance are earnestly solicited.

The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. §1.16 or §1.17 to Deposit Account No. 50-2504. The Examiner is invited to contact the undersigned at 434-972-9988 to discuss any matter regarding this application.

Respectfully submitted,

Michael Haynes PLC

A handwritten signature in black ink, appearing to read "Michael N. Haynes", is written over a rectangular area with a light gray dot grid background.

Date: 7 October 2005

Michael N. Haynes
Registration No. 40,014

1341 Huntersfield Close
Keswick, VA 22947
Telephone: 434-972-9988
Facsimile: 815-550-8850